Table of Contents

Part I SWAN Help 3

Part II Overview 3

1 The program SWAN ... 3
2 Governing equations ... 4
3 Host system requirements ... 5

Part III Model structure 6

1 Discretisation ... 6
2 Data file ... 6
3 Simulation type ... 6
4 Initial conditions ... 7
5 Timestep ... 7
6 Modelled period ... 7
7 Bed resistance ... 7
8 Wetting and drying ... 8

Part IV Boundary conditions 9

1 Prescribe boundary conditions ... 9
2 Boundary types ... 9
Closed boundaries ... 9
Water level boundary ... 10
Discharge boundary ... 10
Weir-type internal boundary ... 10
Topography-based flooding ... 12
3 Boundary value ... 13
4 Typical river boundary conditions ... 15
5 Typical flooding boundary conditions ... 16

Part V Controlling the simulation 17

Part VI Data requirements 18

Part VII Visualise simulation results 19

1 Grid viewer ... 19
2 Colour scale ... 21
3 Caption ... 22
4 Base map ... 22
5 Flow lines ... 23

© 2005 Tamás Krámer, www.vit.bme.hu

6 Flood maps 24

Part VIII Export results 25

1 Export graphics 25
2 Record and export time series 25
3 Export grid data 26
4 Store flow fields 26

Part IX User interface 27

1 Project window 27
2 Simulation window 28
3 Grid view/edit window 29
4 Model parameters dialog 29

Part X Error messages 30

Part XI More information 30

Part XII References 31

Index 0

© 2005 Tamás Krámer, www.vit.bme.hu

1 SWAN Help

SWAN

User's manual

Budapest, 1/24/2006.

This manual describes the use of the two-dimensional flow model SWAN. It is expected that the reader knows the basics of typical WIndows software.

2 Overview

2.1 The program SWAN

SWAN is a graphical Windows software that allows the user to set up, run and post-process the built-in depth-averaged shallow flow model. It describes free-surface, subcritical flow fields by calculating the evolution of the water surface and horizontal depth-integrated velocities. The equations governing the model are the unsteady shallow water equations in Cartesian coordinates, derived by integrating the Reynolds-averaged Navier-Stokes equations in the vertical direction. Losses are calculated by the Manning formula, turbulence and momentum dispersion are estimated using a constant eddy viscosity coefficient.

Development of SWAN

SWAN is developed by Tamás Krámer and János Józsa at the Budapest University of Technology and Economics, Department of Hydraulic and Water Resources Engineering. The original purpose of the model was the simulation of wind-induced flow and sediment dynamics in shallow waters. Later improvements allowed the simulation of river flow and flooding.

Capabilities of the model

- Two-dimensional, unsteady description, on a uniform (possibly stretched) Cartesian grid
- Coverage maps allowa distributed assignment of the Manning-Strickler coefficient
- Constant and isotropic eddy viscosity
- Boundary conditions
  - Prescribe water level and discharge
  - Distribute discharge along the boundary using four different methods
  - Boundary value in simple analytical for or with time series
  - Slip or non-slip closed boundaries
- One-dimensional weirs
- Robust handling of wetting and drying
- Topography-based flooding
  - Fast solution, no stability constraint
  - Distributes water volume with horizontal free surface, without accounting for hydrodynamics
  - Prescribe discharge or water level
- Particle tracking
  - Analytical advective trajectory calculation within a cell
  - User defines seeding points by a pattern or by an explicitly given set of points
- Flood maps
  - Largest extent of the flood
  - Flood arrival time, flood residence time
  - Peak flood level and its time

© 2005 Tamás Krámer, www.vit.bme.hu

- Input and output of simulation results
  - To/from disk, with specified interval
  - Replay a completed simulation
  - „Hotstart": continue a completed simulation
- Standard Windows interface
- Edit, solve and visualise in a single program
  - All parameters are accessible through dialogs
  - Edit grid data in top view and axonometry
  - Velocity field visualisation using arrow field, streamlines, frontlines and particles.
- Manual control of the simulation (start, step, halt)
  - Simulation results can be viewed and copied on the fly, during the calculation
- Data exchange with other applications
  - Input from text and Surfer GRD formats
  - Output to text, Surfer GRD, Surfer BLN and Tecplot formats
  - Clipboard I/O fully supported (allowing e.g. exchange of tabular Excel data)
  - Hotstart from RMA2 finite-element simulation files.
  - Export model to SMS (Surface Water Modeling System).
  - Import picture in WMF format.
  - Save graphical window contents to WMF and BMP format, to file or clipboard

Limitations

SWAN assumes hydrostatic pressure distribution and subcritical flow regime. Its use is not recommended for problems where the horizontal 2D description is inaccurate: processes with scale of the depth or less, in stratified conditions due to density differences and generally in cases where the vertical acceleration significantly affects the Reynolds-averaged flow.

The model's capacity is in general constrained by the available RAM and CPU. The following limits are imposed on the modeller:

- Number of attributes: < 255
- Number of boundary conditions: < 255
- Number of grid nodes: < 10 000 in any direction
- Number of tracked particles: < 10 000

2.2 Governing equations

The motion of free-surface waters is described by the Reynolds-averaged and depth-integrated form of the Navier-Stokes equations, the so-called shallow water equations. This set of PDEs describes the evolution of the flow state variables: namely the water depth (h) and the two Cartesian components of the horizontal volume flux (aka specific discharge). The governing equations take the following form:

$$\frac{\partial h}{\partial t}+\frac{\partial p}{\partial x}+\frac{\partial q}{\partial y}=0$$

$$\frac{\partial p}{\partial t}+\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)+\frac{\partial}{\partial y}\left(\frac{pq}{h}\right)-\frac{\partial}{\partial x}\left(\frac{h}{\rho}T_{xx}\right)-\frac{\partial}{\partial y}\left(\frac{h}{\rho}T_{xy}\right)=$$

$$-\frac{1}{2}gh\frac{\partial(h+z_b)}{\partial x}+fq+\frac{\tau_{sx}-\tau_{bx}}{\rho}$$

$$\frac{\partial q}{\partial t}+\frac{\partial}{\partial x}\left(\frac{pq}{h}\right)+\frac{\partial}{\partial y}\left(\frac{q^2}{h}\right)-\frac{\partial}{\partial x}\left(\frac{h}{\rho}T_{yx}\right)-\frac{\partial}{\partial y}\left(\frac{h}{\rho}T_{yy}\right)=$$

$$-\frac{1}{2}gh\frac{\partial(h+z_b)}{\partial y}-fp+\frac{\tau_{sy}-\tau_{by}}{\rho},$$

where the bottom shear stress is calculated by the Manning formula:

© 2005 Tamás Krámer, www.vit.bme.hu

$$\tau_{bx} = \frac{\rho g}{k^2 h^{7/3}} \sqrt{p^2 + q^2}\, p,$$
$$\tau_{by} = \frac{\rho g}{k^2 h^{7/3}} \sqrt{p^2 + q^2}\, q$$

the elements of the depth-averaged effective turbulent stress tensor are obtained using

$$T_{xx} = 2\rho\nu_e \frac{\partial u}{\partial x}, \quad T_{xy} = T_{yx} = \rho\nu_e \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right), \quad T_{yy} = 2\rho\nu_e \frac{\partial v}{\partial y}$$

The symbols used in the above equations are the following:

x, y	= horizontal Cartesian coordinates [m],
t	= time [s],
p, q	= x and y components of the unit discharge **q** [m^2/s],
h	= water depth [m],
u, v	= x and y components of the depth-averaged velocity **v** [m/s],
z_b	= terrain elevation [m],
η	= water surface elevation [m], ($\eta = z_b + h$),
g	= acceleration due to gravity [m/s^2],
$\nu_{e,H}$	= effective horizontal eddy viscosity coefficient [m^2/s],
τ_{bx}, τ_{by}	= x and y components of the bottom shear stress [Pa],
τ_{sx}, τ_{sy}	= x and y components of the surface shear stress (due to wind) [Pa],
ρ	= water density [kg/m^3],
k_{Str}	= Strickler terrain smoothness [$m^{1/3}/s$].

The governing equations in SWAN are in the so-called nonconservative form, which assumes that the solution is continuous therefore it is only applicable to subcritical flow. More about the model can be found in the paper by Józsa et al. (1999). The shallow water equations are described in detail in the book by Kozák (1977).

Numerical solution

The solution is obtained numerically in finite timesteps, on an equidistant Cartesian grid using the finite difference method. Spatial derivates are approximated by second-order central differences. Time marching is accomplished with an explicit first-order Euler or second-order Runge-Kutta scheme. Wetting-and-drying is handled by excluding very shallow cells momentarily from the calculation and by gradually switching of terms in the momentum equation vanishing with depth.

2.3 Host system requirements

Hardware requirements

The program runs on all tested PC's equipped with 32-bit processors. Simulations on large grids (>40 000 nodes) may be so demanding that concurrent applications slow down considerably. All data (including inactive grids in the file) must simultaneously fit in the memory, but the explicit nature of the solver means that no coefficient matrix must be built, therefore the RAM in today's (2005) desktop computers is never a limiting factor. If the simulation is written to disk incrementally, the written files typically use several hundred megabytes. The application consists of one executable of less than 1 megabyte.

Software environment

SWAN requires any Microsoft Windows operating system newer than Windows 95. Due to the limitation of the compiler, the code is 16-bit, so it works only with old DOS-style folder and file names.

How to install and uninstall?

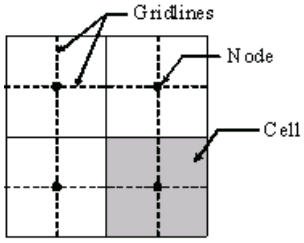
Copy files SWAN.EXE and CTL3D.DLL to the desired destination folder. The software is removed by simply deleting these files.

© 2005 Tamás Krámer, www.vit.bme.hu

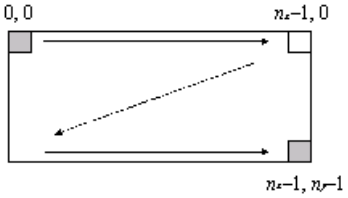
3 Model structure

3.1 Discretisation

For the numerical solution, the flow domain is split to identical rectangular cells. We use the regular grid defined by the cell centres to discretise the domain. The nodes of the grid are in fact equivalent to the cell centres.



The digital elevation model and the attribute map are in the same resolution as the computation. The grid nodes are indexed so that the row index runs faster than the column index. Indexes range from 0 to n–1, where $n = (n_x,n_y)$ is the number of cells in the given direction.



3.2 Data file

Except optional hotstart and replay files, all input data of a simulation is contained in a single project file with extension SWN. The file is binary and can only be edited within SWAN. Besides that, the software also provides text-based export/import for grids and data series.

The simulation results are not stored in the SWN file. Those results are lost unless we make sure from the beginning that they are written to output files (see Result output).

3.3 Simulation type

The model solves the unsteady governing equations, and in addition allows steady-state simulations.

Steady-state simulations

We let the unsteady solution converge to the steady-state solution by holding all boundary conditions constant. Theoretically convergence takes extremely many iteration steps to reach machine accuracy. Practical accuracy requirements are generally more relaxed and permit shorter simulations. Since SWAN does not provide an automatic convergence stopping criterion, the user must monitor the simulation and stop it either manually or by pre-programming the end time of the simulation. It is generally adequate to judge the level of convergence by following the evolution of a representative nodal water surface or velocity value in a grid viewer window or by recording the time series.

Unsteady simulations

By default SWAN yields an unsteady solution. In contrast to steady-state simulations, the inaccuracy of initial conditions affects the first stage of the simulation.

© 2005 Tamás Krámer, www.vit.bme.hu

3.4 Initial conditions

We must prescribe the initial water level and volume flux at each grid node.

Cold start

With a "cold start", the simulation starts with water at rest: all wettable cells are filled up to the user-defined initial water level and velocities are set to nil. To set the initial water surface, choose **Definitions/ Model parameters** from the menu and edit the corresponding field in the dialog box. Cells in which the bottom elevation is above the initial water level will become dry.

We can set a variety of initial conditions using boundary conditions supplemented with the *InitOnly* switch. To fill ponds individually by controling water level or volume, use the topography-based flooding facility.

Hot start

"Hot starting" means that initial conditions are supplied by the final result of a previous simulation. This is especially convenient if the siimulation is to be repeated with a slight change in the parameters. In SWAN, hot-starting is accomplished by reading stored simulations from disk.

3.5 Timestep

The timestep is selected not only according to the desired time resolution but also considering stability and accuracy constraints. Variations of the flow or boundary conditions that are faster than the timestep are described only in an average sense. The so-called CFL (*Courant-Friedrich-Lewy*) stability crierion imposes an upper limit to the timestep of the explicit time marching scheme:

$$\Delta t \le \min_{i,j} \frac{\alpha_{CFL}}{\frac{\sqrt{g h_{i,j}} + u_{i,j}}{\Delta x} + \frac{\sqrt{g h_{i,j}} + v_{i,j}}{\Delta y}}$$

where the minimum is taken over all cells. Numerical instability manifests itself in excessively oscillating solution, negative depths and unrealistically high velocities. The stability of the solution may require that the timestep dictated by the CFL criterion is reduced by a factor α_{CFL} = 0.5…0.95. If we enter a value for α_{CFL}, then SWAN estimates Δt with the above formula in each timestep. If only Δt is specified and the field α_{CFL} is left blank, then the timestep is kept constant and the model does not check whether it fulfils the CFL-criterion.

Example: In a flood application, the minimal value of Δt was found near the inflow boundary, with Δx = 100 m; g = 9.81 m/s^2; h = 3 m; u = 1.5 m/s, the allowed timestep is approximately

$$\Delta t \le \frac{100}{\frac{\sqrt{9{,}81 \cdot 3} + 1{,}5}} = 14{,}4\,s$$

3.6 Modelled period

The start time of the simulation is given with an actual date and time. If the end time is entered, the simulation is paused automatically after reaching it, otherwise it must be stopped manually. The advantage of using actual dates and times is that modelled flow parameters can be compared directly with measurements. Of course the absolute times have no significance with steady-state simulations.

Dates are given according to the Regional settings of Windows, also shown in the dialog box (for example m/d/y means month/day/year, i.e. Dec 24, 2005 is entered as 12/24/05)

3.7 Bed resistance

Bed resistance is calculated by the Manning formula:

$$\boldsymbol{\tau}_b = \frac{\rho g}{k^2 h^{1/3}} |\mathbf{p}| \mathbf{p},$$

© 2005 Tamás Krámer, www.vit.bme.hu

where k [$m^{1/3}/s$] is the Strickler smoothness coefficient (the inverse of the Manning roughness coefficient, $k = 1/n$). SWAN takes into account the distribution of k according to the land use, vegetation type and bed characteristics. The Manning formula is used to describe resistance due to both surface roughness and vegetation, with the assumption that k is independent of the local water depth.

Attributes

The Strickler smoothness coefficient is not directly specified on the grid. Instead, the coverage is classified into a limited number of coverage types and an *attribute definition* is created for each type. These attributes describe the value of k, and whether the cell is accessible to water. A new project defines by default two types of attributes: "Land" and "Water" and assigns the ID 0 and 1 to them, respectively. We can add further attribute types, which are automatically numbered with a unique ID. Some software refer to attribute as "material type".

Attribute map

In the end, the attribute ID is assigned to each grid cell through the *attribute map*. This map is in fact a grid of the same size as the elevation grid but contains IDs that refer to the corresponding attribute. The attribute map can be edited graphically within SWAN or exchanged with other software (such as Excel) as an ASCII table. A SWAN project file may contain more than one attribute maps, in which case we must activate the one that is used in the simulation.

In a typical river model, the main riverbed and the floodplain are distinguished by their own attribute. For further differentiation, the cleared, cultivated and forested floodplain can all have their own attribute. Seasonal variability of the hydraulic resistance is then implemented by changing k in the attribute definitions. To exclude cells from the flow domain, their attribute ID is set to 0 on the attribute map. Or, if all cells with a given attribute are to be excluded, the definition of that attribute is modified to *inaccessible to water*.

3.8 Wetting and drying

Following general practice, cells are considered dry if the depth is less than a *truncation depth*, i.e. $h < h_{trunc}$, otherwise they are wet. Dry cells are excluded from the flow computation in that timestep.
To increase robustness of the solution, two intermediate depths are introduced (h_{min}, h_{crit}), by which the terms of the momentum equations sensitive to shallowness are gradually switched off with a factor α (Burchard 1998):

$$\alpha = \frac{h - h_{min}}{h_{crit} - h_{min}}, \text{ if } h < h_{crit}.$$

The value of h_{min} and h_{crit} is problem specific, generally in the order of 0.01 to 0.1 m. To set their value, select **Definitions/ Flow model** in the menu, choose *Method* = *Burchard* and click on *Wetting and drying/Setup...* The meaning of the fields:

- *Minimal depth*
  = Value of h_{min}
- *Critical depth*
  = Value of h_{crit}
- *Truncate depth*
  = Value of h_{trunc}. A cell is considered dry if its depth is smaller than that. To have at least this deep water in all cells, swith on the *Truncate very small depth values* flag.
- *Additional viscosity*
  Experimental. Should be set to 0.

The following relationship must be respected : $0 < h_{trunc} <= h_{min} <= h_{crit}$.

© 2005 Tamás Krámer, www.vit.bme.hu

4 Boundary conditions

4.1 Prescribe boundary conditions

Along open boundaries of the flow domain, the solution of the governing equations must be extended by boundary conditions that prescribe water level or discharge. The shoreline constitutes a closed boundary, where normal flow is not permitted.

The flow state (h, p, q) set at open boundaries may given as a constant, an analytically function or a time series.

To prescribe boundary conditions (BC) in SWAN, we must first define the list of boundary conditions, then assign those conditions to cells that make up the open boundary.

List of boundary conditions

Select **Define/Boundaries** to access the dialog where boundary conditions (BC) can be defined. The following typical situations arise:

- River sections: the inflow and the outflow BC must be supplied.
- Flooded terrain: a BC is supplied for each levee breach
- Bays: a BC is supplied for the open boundary at the entrance of the bay
- Lakes, reservoirs: closed boundaries are automatically handled by the model

Boundary conditions added to the list are assigned an ID, which is used to mark cells where that BC is active.

Boundary map

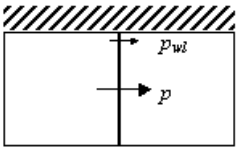
To relate boundary conditions to the computational grid, we use the *boundary map*. The boundary map is a grid that contains the IDs (integer numbers between 0–255) identifying the boundary condition. No more than one BC can be prescribed in a cell and the value 0 is reserved for cells where no boundary condition is prescribed (typically the boundary map is 0 in most of the domain). To edit the boundary map, open a grid editing window in SWAN or use another application to edit the tabular map and paste it into SWAN through the clipboard. It is important that BCs must be defined prior to entering their IDs in the boundary map. A SWAN file may contain several such boundary maps, in which case the one to be used in the simulations must be activated. The simulation must be restarted if the boundary map is altered.

4.2 Boundary types

4.2.1 Closed boundaries

At closed boundaries (i.e. levees, shoreline) the velocity components perpendicular to the boundary will automatically be set to 0,so that the user does not have to define any boundary condition here. The discharge component parallel to the closed boundary (p_{wl}) is also determined by the model: $p_{wl} = A\ p$, where p is the closest specific discharge inside the domain (see figure below) and A is a wall slip coefficient. The value of A is any number between 0–1:

- A = 0: lateral flow is reduced to 0,
- A = 1: lateral flow is unaffected by the wall.



The value of A is given universally for all closed boundaries, by choosing the command **Define/Flow** from the menu and filling the **wall slip coeff.** field.

© 2005 Tamás Krámer, www.vit.bme.hu

4.2.2 Water level boundary

The absolute water level (and not the water depth) is prescribed in the center of the cells.

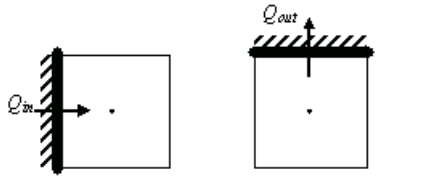
4.2.3 Discharge boundary

At these open boundaries, the total discharge Q is prescribed. The model then distributes the discharge among the boundary cells by keeping $(vh)_i$, v_i, $(v/h)_i$ or $(S_M)_i$ uniform, where v_i = the velocity component normal to the boundary, h_i = local water depth, $(S_M)_i$ = the energy slope computed using the Manning formula, $S_M = v^2/(k^2 h^{4/3})$.

distribution of Q	**weight of q_i**	**weight of v_i**
a) vh = const	1	h_i
b) v = const	$1/h_i$	1
c) v/h = const	$1/h_i^2$	$1/h_i$
d) S_M = const	$1/(kh_i^{5/3})$	$1/(kh_i^{2/3})$

Option (a) is equivalent to setting the specific discharge $q_i = Q/L$, where Q is the total discharge, L is the boundary length. If the depth or the smoothness has a varied distribution along the open boundary, option (d) may give the most realistic steady-state approximation. It is expected that the error made by an inaccurate distribution of Q is reduced with distance from the boundary because the governing equations rearrange the discharge according to the actual hydraulic situation. It is therefore advised to set up the open boundaries at locations with predictable (close to uniform) flow distribution, extending by some distance the limits of the studied domain.

The internal layout of the computational grid is such that the components of the specific discharge lie at the midpoint of the perpendicular cell side, not at the cell center (this is the so-called *staggered* layout). As the boundary conditions are assigned to cells (through the boundary map), the open side of the boundary cells must be specified explicitly in the definition of the boundary condition. The open side is adjacent to an external and an internal cell; we must assign the BC to the internal cell. Finally, we must specify whether positive discharges mean outflow or inflow. Two exampes are shown for discharge boundaries and their definition.



4.2.4 Weir-type internal boundary

Flow over weirs and levees undergoes large acceleration, so the assumption of hydrostatic pressure conditions introduces error in the solution. At these places, the stage-discharge relationship used for weirs is more accurate than the shallow water momentum equations. The weir formula yields the specific discharge in function of the upstream water level, weir crest elevation and downstream water level:

$$q_w = \beta \cdot C_w \cdot \frac{b}{\Delta n}\left(z_{s,u} - z_w\right)^{1.5}$$

where:

© 2005 Tamás Krámer, www.vit.bme.hu

β	= submergence factor (see below)
b	= weir crest width (across flow)
Δn	= grid interval across flow
$z_{s,u}$	= upstream water surface elevation
z_w	= weir crest elevation
C_w	= weir flow coefficient (C_w = 1,6–1,7)

C_w may be expressed by the discharge coefficient μ (used in some weir formulations) as

$$\mu = \frac{3}{2\sqrt{2g}} C_w$$

The effect of submergence caused by tailwater above the crest elevation is accounted for by the following correction factor, proposed for broad crestesd weirs:

$$\beta \begin{cases} = 1.0 & (\gamma \le 0{,}67) \\ = 1.0 - 27.8(\gamma - 0{,}67)^3 & (\gamma > 0{,}67) \end{cases}$$

in which $\gamma = (z_{s,u} - z_w) / (z_{s,d} - z_w)$ and $z_{s,d}$ is the downstream surface elevation.

When submergence occurs, the model takes the linear combination of the flow determined by the weir formula (q_w) and the solution of the shallow water equations (q_{SWE}):

$$q = \beta q_w + (1 - \beta) q_{SWE}.$$

Consequently, free overflow ($z_{s,d} < z_w$) is calculated purely by the weir fomula, whereas above submergence corresponding to γ = 0,67, the shallow water solution dominates more and more. If the surface is lower than the weir crest on both sides, the weir will inhibit any flow, like an impervious wall.

Weirs are implemented in SWAN as internal boundary conditions of discharge type defined by weir-specific auxiliary data. No boundary value is directly given. The row

```
Weir
```

must be included in the auxiliary data, which tells the model that the discharge is instead calculated by the weir formula according to the previously detailed procedure. The optional line

```
WeirCoeff = 1.7
```

overrides the default weir coefficient C_w = 1.6 $m^{0,5}/s$. If the weir width is not equal to the cell size, the actual width (150 m in this example) is specified with the line

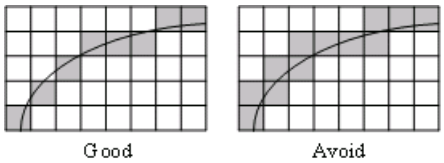
```
WeirWidth = 150
```

When no width is given, SWAN takes b equal to the cell size.

The crest level may be specified in two ways: through the topography grid or using the auxiliary data.

The more common use is that the model takes z_w from the topography grid so that levees with variable elevation can be described. In the shallow water solution q_{SWE}, the model calculates with the average terrain elevation of the adjacent cells, instead of the weir elevation, as z_w is generally representative only to a small portion of the cell. We note that the topography grid listed in the model window is already the modified version of the input topography grid, where the elevation in weir cells were overridden by the average elevation of adjacent non-weir cells. When crest elevation is given through the topography grid, SWAN applies the weir formula on the downstream side of the weir cells, disregarding the the boundary side specified in the definition of the boundary condition. This procedure ensures that the weir functions as a closed wall while the surface elevation in adjacent cells is below the crest elevation. As the following figure shows it, the impervious effect is already created by diagonally connected weir cells. The curve designates the approximated weir centerline, cells shaded with gray are those marked as boundary cells in the boundary map.

© 2005 Tamás Krámer, www.vit.bme.hu



Grid-aligned weirs with horizontal crest are more conveniently specified using the auxiliary data. We must include the line

```
WeirLevel = 146.12
```

where the value stands for the absolute elevation of the weir crest in m. Unlike in the previous method, the active topography grid must initially contain the surrounding average terrain elevation in the weir cell, not the crest elevation. With this method, the weir formula is applied only to the cell side defined in the boundary definition, the same side for all weir cells of that boundary ID. That is why it is advisable to employ the *WeirLevel* line for grid-aligned weirs only.

The crest elevation or the weir width may change in time. To prescribe time-dependent b or z_w, specify the name of an existing time series (added to the project window as scalar time series) which contains the time variation of $b(t)$ or $z_w(t)$. Unlike other boundary types, this is not selected from the list of time series in the boundary definition window (which would define $Q(t)$ directly). Example: the lines

```
WeirLevel = "Z_w(t)"
WeirWidth = "B_w(t)"
```

take the crest level and width from the time series named Z_w(t) and B_w(t), respectively. A restriction is that the name of the time series must letters other than digits, otherwise the time series named "12.3" wil be recognised as the constant value 12.3, for example.

4.2.5 Topography-based flooding

Any water level boundary can be extended to cells surrounding the boundary by including the line *FloodFill* in the auxiliary data with one of [*ByWLevel*, *ByVolume, ByDischarge*] as parameter.

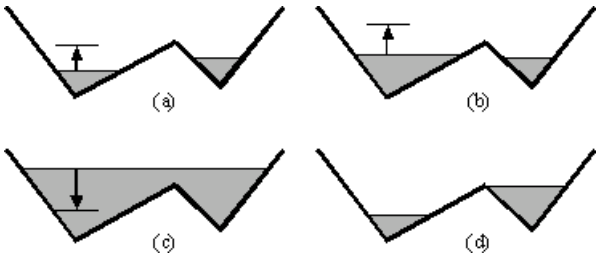
With the line

```
FloodFill=ByWLevel
```

all cells connected to the boundary will be flooded to the given water level if their elevation is lower. Cells separated from the boundary by a ridge higher than the prescribed water level are not affected. Flooding to a certain water level has two primary uses:

- combined with the *InitOnly* switch, it sets the initial water level in topographic depressions independently from the global initial water level
- if the prescribed water level is time dependent, it becomes the instrument of topography-based flooding.

The next series of figures illustrates the working of the flood algorithm. The boundary condition imposes a time-dependent water level in the cell under the arrow, located at the bottom of a left pit. Thanks to the *FloodFill* switch in the auxiliary data, the whole pit is filled up to this level, however the right pit is flooded only after the separating ridge is exceeded. Furthermore, the subsequent lowering of the water level leaves the right pit flooded to the level of the ridge. In all cases, the velocity components are set to zero.



© 2005 Tamás Krámer, www.vit.bme.hu

Specifying *ByVolume* instead of *ByWLevel* in the auxiliary data changes the water level until the prescribed water volume has been spread around the boundary. We obtain a piecewise discontinuous water surface in general. When a ridge is exceeded, the model follows the steepest descent from the lowest point of the ridge to find the next recipient water body behind the ridge where the distribution can continue. Even though the

```
FloodFill=ByVolume
```

boundary is of water level type, the boundary value is exceptionally interpreted as the water volume to be distributed, in m^3. This can be a negative number to specify extraction.

A variant of topography-based flooding by volume is topography-based flooding by discharge. This is specified with the follwing line in the auxiliary data field:

```
FloodFill=ByDischarge
```

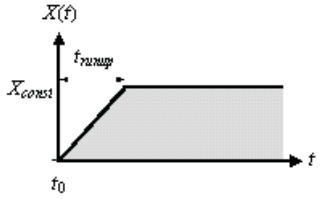
With that line, the procedure is the same as with the *ByVolume* parameter, but the volume increment (or decrement) will be timestep dependent. The boundary value is the discharge in that case, in m^3/s, though this is not reflected in the dialog window. It is emphasised that the two latter applications (*ByVolume* and *ByDischarge*) also require a boundary condition of water level type like *ByWLevel*.

4.3 Boundary value

The water level or discharge imposed at the boundary may be constant or unsteady. Unsteady boundary values (denoted in the following by $X(t)$) are either periodic or specified by time series. When stepping from time level t to $t+\Delta t$, the model actually uses the average value of $X(t)$ during the interval $[t, t+\Delta t]$.

Constant boundary value

For constant $X(t)$, the steady value X_{const} and an optional runup (ramping) of duration t_{runup} is given.

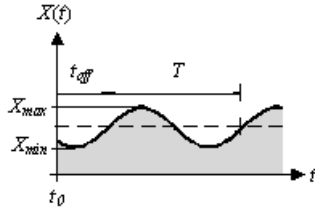


Periodic boundary value

The periodic value is generated by the harmonic function

$$X(t) = \frac{X_{max} + X_{min}}{2} + \frac{X_{max} - X_{min}}{2} \sin\left(\frac{t - t_{off}}{T}\right)$$

ahol t = time since model start [s], substituted by the solver when evaluating the function; X_{min}, X_{max} = minimum and maximum value; t_{off} = phase offset [s]; T = period [s]. The last four parameters are supplied by the user.

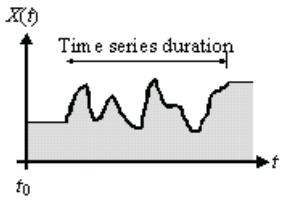


Value given with time series

Any variation of the boundary value can be prescribed using time series with uniform time resolution. The time series must be first imported into the project window with Scalar type. The absolute model time must be in accordance with the interval of the time series, e.g. a time series for the period March 1-10, 2000 will provide boundary values for a 5-day simulation starting with a model time of March 5,

© 2005 Tamás Krámer, www.vit.bme.hu

2000. As the next figure illustrates, if the time series is queried for a time beyond its time interval, the first or last value is returned.



Auxiliary data

We can provide auxiliary data to extend the functionality of basic (water level and discharge) boundary conditions. The auxiliary data field contains keywords and their values separated by an equal sign and any number of spaces

```
[Keyword]=[Value]
```

The keyword can be a *switch*, in which case the value is 1 or 0 in order to switch it on and off, respectively. If only the keyword is specified, it is switched on.

In some cases, the auxiliary data fully define the boundary condition and no boundary value needs to be explicitly defined with the constant, periodic or time series fields. The list table gives the complete list of keywords by category available in the auxiliary data field.

For any boundary types:

- *InitOnly*
  The boundary condition is in fact an initial condition, applied only when initialising the model.
- *ZeroCurv*
  The boundary is obtained by extrapolation from the interior so that the second derivative is 0. For a boundary on e.g. the east side, this gives $X(t, i) = 2X(t, i–1) – X(t, i–2)$. This method is valid only for cells adjacent to the outer boundary of the domain, otherwise an error message is shown. The boundary value is therefore given purely by the auxiliary data.
- *ZeroCurvIfOut*
  Overrides any explicitly given boundary value $X(t)$ with the *ZeroCurv* zero-curvature extrapolation if the flow points out of the flow domain. Like *ZeroCurv*, this keyword is valid only for cells adjacent to the outer boundary. .

For water level boundaries:

- *FloodFill*
  Achieves the desired water level or volume change by filling or draining the surrounding cells with piecewise horizontal water surface. The value *ByWLevel, ByVolume* or *ByDischarge* is required.
- *AreaSource*
  A global source (or sink): evaporation, infiltration (negative value) or rainfall (positive value). The rate of evaporation/ infiltration/ rainfall is given by the boundary value, the unit being not [m], but [mm/d] in this special case. If this boundary condition is defined and enabled, it is applied over the whole domain without the need for representation on the boundary map.

For discharge boundaries:

- *NoCrossVel*
  The velocity component perpendicular to the prescribed q is set to 0. E.g. if the position of the boundary is the eastern side, the vertical velocity components on the N and S sides are set to 0.
- *NoCrossVelIfOut*
  Similar to the *NoCrossVel* switch except that the transverse velocity components are set to 0 only if the actual flow is outward (which happens with positive outflow or negative inflow).
- *Weir*
  This signals an internal boundary condition, where the specific discharge is determined with a weir formula. It is specified only with auxiliary data.

© 2005 Tamás Krámer, www.vit.bme.hu

- *WeirLevel*
  Weir crest elevation, [m]. Optional. Used only when the *Weir* switch is also defined.
- *WeirCoeff*
  Weir coefficient, [$m^{0.5}$/s]. Optional. Used only when the *Weir* switch is also defined.
- *WeirWidth*
  Weir width, [m]. Optional. Used only when the *Weir* switch is also defined.

The details of each auxiliary data keyword is summarised in the following table.

Keyword	***z*-bnd**	***Q*-bnd**	**Value**	**Default**	***X*(*t*) value**
AreaSource	yes	–	switch	0	$S(t)$, [mm/d]
FloodFill	yes	–	*ByWLevel* *ByVolume* *ByDischarge*	ByWLevel	$z_s(t)$, [m] $V(t)$, [m^3] $Q(t)$, [m^3/s]
InitOnly	yes	yes	switch	0	anything
NoCrossVel	–	yes	switch	0	anything
NoCrossVelIfOut	–	yes	switch	0	anything
Weir	–	yes	switch	0	–
WeirCoeff	–	yes	C_w	1.6 $m^{0.5}$/s	–
WeirLevel	–	yes	z_w	$z_b(i, j)$	–
WeirWidth	–	yes	b	Δx, Δy	–
ZeroCurv	yes	yes	switch	0	–
ZeroCurvIfOut	yes	yes	switch	0	–

4.4 Typical river boundary conditions

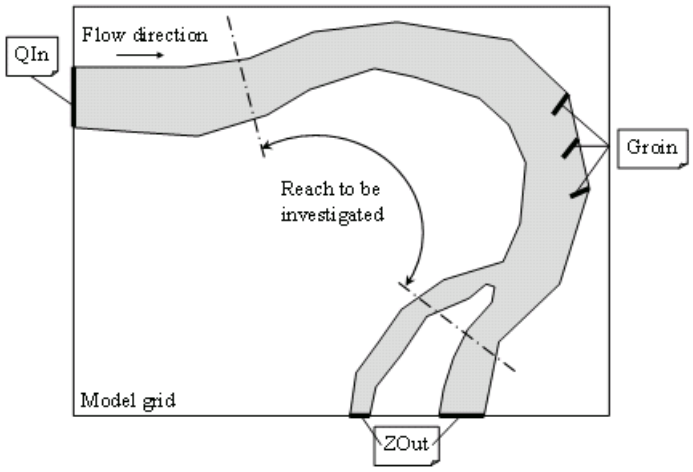
To simulate steady-state flow on river sections, a discharge boundary (Q_{in}) is set up at the inflow boundary and a water level boundary (Z_{Ki}) is set up at the outflow boundary. The initial water surface elevation is set high enough to submerge both the inflow and outflow boundaries. The study area is extended by a distance of several river widths both in the upstream and downstream direction so as to reduce boundary effects. The boundaries should possibly lie in well defined sections with more or less uniform velocity distribution.

The boundary values are constant for the steady-state simulation. To minimise the generation of artificial waves due to the imperfect initial conditions, the discharge increases gradually by specifying a runup of at least one hour. As an example, the discharge starts at 0 and reaches its final value of 1000 m^3/s in 1 h. Meanwhile the outflow water level decreases from 50,0 m to 48,5 in 1 h. SInce the built in analytical functions do not provide such a variation, the time series {50; 48.5} is imported with one-hour interval and set as boundary value for the water level. To improve stability at the outflow boundary, we enter *NoCrossVel* in the discharge auxiliary data of the outflow boundary condition.

Any groins are modelled as weirs. Since they are inclined relative to the gridlines, they must be represented with stepped cells. The cell side cannot be prescribed uniformly therefore the weir crest elevation must be given through the topography.

The river banks constitute a closed boundary that is handled automatically by the model without any explicit boundary condition.

© 2005 Tamás Krámer, www.vit.bme.hu



- Boundary Q_{In}
  Water level: Nothing is prescribed.
  Discharge: W side, distribute by keeping S_M = const, inflow is positive, prescribe, value is constant Q = 1000 (m^3/s), runup = 3600 s.
- Boundary Z_{Out}
  Water level: Prescribe, given by time series (and the previously created {50; 48.5} timse series is selected in the list box).
  Discharge: S side, outflow is positive, prescribe, value by auxiliary data = „*NoCrossVel, ZeroCurvature*".
- Boundary *Groin*
  Water level: Nothing is prescribed
  Discharge: Prescribe, value by auxiliary data = „*Weir*".

4.5 Typical flooding boundary conditions

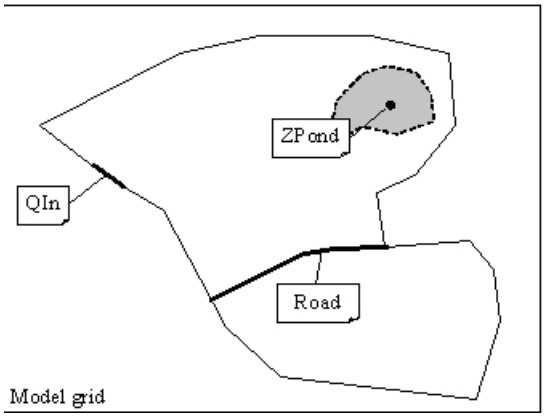
The example shown here refers to the modelling of an initially dry floodplain that is flooded following a levee breach. The extent of the flooded area changes during the unsteady simulation. No specific boundary condition must be specified for the wave front, the model handles it automatically like closed boundaries.

The lake situated on the floodplain is filled initially using a water level boundary. We define a condition with topography-based flooding (*FloodFill*) so that its is sufficient to assign the boundary condition to one single cell on the boundary map. The remaining floodplain is initially dry, therefore the initial water level is in the model parameters set to a value lower than any terrain elevation.

The discharge enters the floodplain through the breach (boundary Q_{in}), that is determined by a one-dimensional river model or a time series. As the boundary is closer to the horizontal gridlines, the discharge is positioned on the S cell sides regardless of the stepped representation of the boundary. The total discharge is distributed by keeping (q = const). Any other choice would cause an error because the depth is initially zero along the boundary.

The area is crossed by a road, whose embankment is modelled as a weir. Not only the road is inclined relative to the gridlines but it also has variable elevation, consequently the weir crest elevation must be specified through the topography grid.

© 2005 Tamás Krámer, www.vit.bme.hu



- Boundary Q_{in}
  Water level: Nothing is prescribed.
  Discharge: S side, distribution q = const, inflow is positive, prescribe, given by time series.
- Boundary Z_{lake}
  Water level: Prescribe, constant value = 57,4 (m), runup = 0, auxiliary data = „*InitOnly; FloodFill = ByWLevel*".
  Discharge: Nothing is prescribed.
- Boundary *Road*
  Water level: Nothing is prescribed.
  Discharge: Prescribe, (position and distribution is indifferent), given by auxiliary data = „*Weir*".

5 Controlling the simulation

Before starting the simulation set all model parameters, select the active topography, attribute map and boundary map. To begin the simulation, open the simulation window, which initialises internal variables and sets the model to the initial state. Once the simulation window is open, the simulation can be started, paused, stepped manually or reset to the initial state using the toolbar buttons or the menu. The project window may be closed at the end of the simulation.

Initialise

With the initialisation (**Calculate/Initialise**) the model allocates internal structures in memory and fills them according to the initial conditions. This step opens the *simulation window* appear on the right which lists the input and output grids and time series. The active input grids should not be resized while the simulation window is open.

Start

The command **Calculate/Start** launches the simulation. If it is grayed, then the model must be first initialised.

Step

The command **Calculate/Step** advances the simulation with a single timestep then stops.

Stop

The simulation can be stopped (paused) by the command **Calculate/Stop** or the F8 key. After the completion of each timestep, the execution is returned to Windows. If we open a dialog window while the simulation is running in the background, the simulation is automatically suspended till the dialog is closed.

SWAN will propose to finish the simulation when the *End time* given with the model parameters has been reached. If that field is left blank, the simulation can be stopped only manually.

Closing the model window stops the simulation, closes output grid viewers and frees memory allocated to internal structures. To put away the simulation window without terminating the simulation, minimise

© 2005 Tamás Krámer, www.vit.bme.hu

the window instead of closing it.

Reset

The command **Calculate/Reset** rewinds the simulation to the initial state. The latest results will be lost, but the settings of open grid viewers and recorded time series are preserved. Most model parameters may be changed on the fly, some changes however require an immediate Reset:

- Editing the active topography, attribute map and boundary map,
- Switching the "Accessible to water" property of an active attribute.
- Changing the type of an active boundary condition (water level to discharge, or the other way),
- Switching the Read/write simulation results,
- Requesting flooding statistics.

Simulation stops with an error

The simulation can also stop due to numerical overflow caused for example by division with a very small number. To have the model stop with an error message instead of program termination in such cases, switch on the checking of water depth and discharge with the command **Define/Model parameters** then the button **Overflow handling**.

6 Data requirements

Digital elevation model

A digital elevation model (DEM) based on a regular grid is necessary to represent the topography. The values it contains are the average elevation of the cell above a given datum, in metres. The resolution of the DEM defines at the same time the resolution of the simulation. The grid must not necessarily be oriented north, but the cells sides are named according to the default orientation when defining the discharge boundary position.

An interpolation technique is generally necessary to create the DEM. One must generally try to ensure that the elevation points cover the whole domain with the appropriate density, and extrapolation is best avoided. For a smooth surface and uniformly distributed elevation points, the kriging interpolation generally yields good results, otherwise linear interpolation with triangulation is recommended. The elevation of cells inaccessible to water (i.e. exterior cells) plays no role in the calculations, it influences only the graphical appearance in the grid viewer.

SWAN accepts the DEM as an ASCII table with the elevation values listed from the top-left to the bottom-right corner, separated with tab or space and rows ending with a carriage return. Decimal points must be used, not decimal points. Surfer GRD files are also accepted in any flavour (ASCII, binary or Surfer 7).

Attribute map

The spatial distribution of the hydraulic resistance and the extent of the flow domain is defined by the attribute map. It is generally produced from topographic maps, land use maps or aerial photographs. The terrain is classified from the perpective of the hydraulic properties (surface characteristics, vegetation type...) and the attributes are defined for these classes. Create a vector drawing of the boundaries and rasterise this vector drawing in the resolution of the DEM.

Like the DEM, the attribute map is imported as an ASCII table or a Surfer GRD file. The attributes are referred to by their ID in the range 0–255.

Time series

Unsteady boundary conditions are specified using time series imported into the project file. These time series consist of scalar values (water level, discharge, evaporation, weir width etc.) sampled at constant interval.

Time series are imported as ASCII files, organised in a column. The timestep and the start time is set in a dialog box.

© 2005 Tamás Krámer, www.vit.bme.hu

Base map

The flow solver does not need it, but it can be used to enhance the grid viewer with a background map. SWAN accepts Windows metafiles (WMF). A WMF file is given in drawing coordinates, therefore the geographic coordinates of the bounding rectangle must be specified graphically in the Picture properties dialog box.

7 Visualise simulation results

7.1 Grid viewer

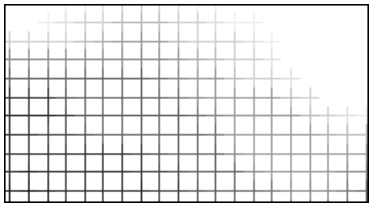
Fields represented on grids are visualised in plan or axonometric view in the grid viewer window. SWAN offers numerous graphical elements to present the two-dimensional information with eye-catching graphics, nevertheless suitable for technical analysis. These elements include color scales, contours, vector fields and particle traces. Any number of grid viewer windows may be open simulaneously, each of them configured individually.

Many software (such as Excel, Matlab etc.) are available to visualise time series, therefore no such possibility is offered in this version of SWAN. Instead, time series data are quickly exported to other applications through text files or the clipboard.

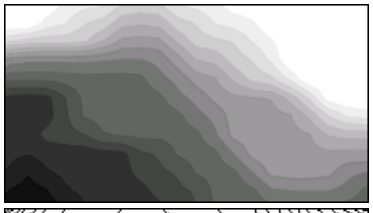
The grid viewer is updated automatically if the grid content changes, which is useful for tracking the evolution of the simulation.

Four main graphical elements can be laid on top of each other:

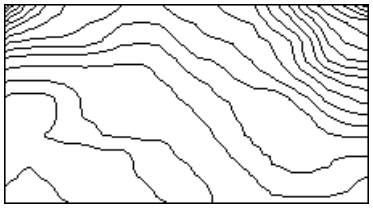
1. G*ridlines* that connect cell centres (which are not equivalent to cell boundaries).
2. A *fill* shades cells according to the variable value.
3. *Contours* are drawn independently of the pattern, with the specified interval and colour coding.
4. Vector fields are visualised by an *arrow* field of arbitrary magnification and distribution.



Gridlines

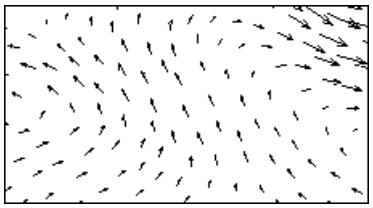


Fill



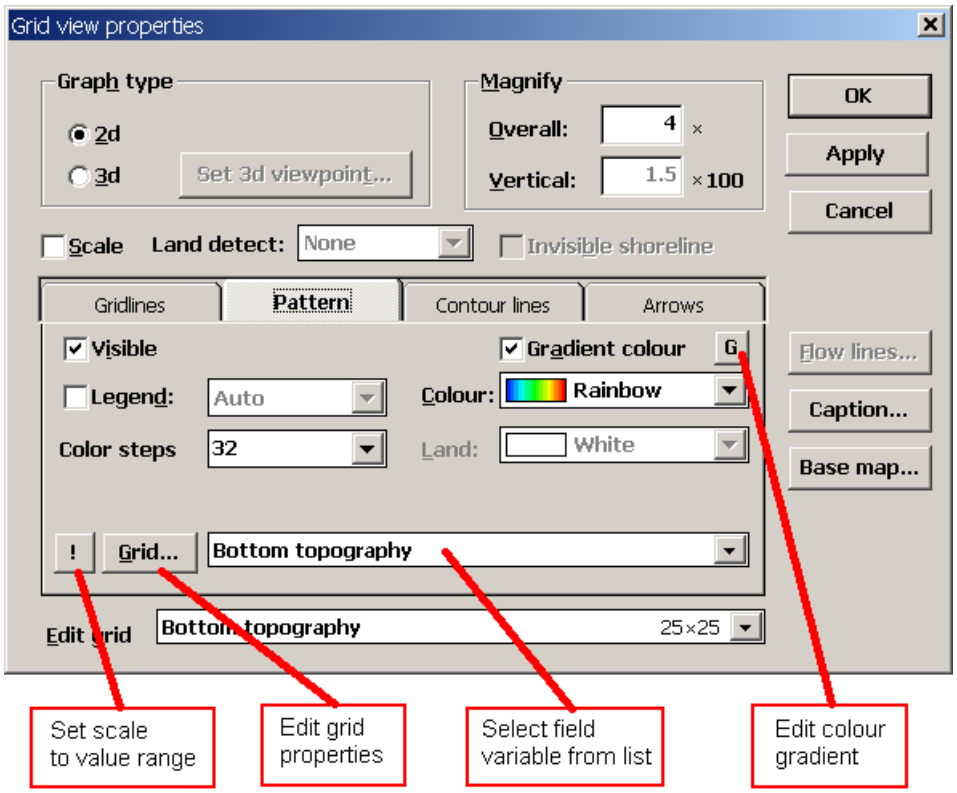
Contours

© 2005 Tamás Krámer, www.vit.bme.hu


Arrows

The axonometric view offers only a subset of the plan view. It does not draw vectors, traced particles and background map.

To configure the grid view window right-click or select **View/ Grid view properties...** from the menu. You can also access five zoom factors from the **View** menu, but the zoom (magnification) factor can be set numerically in the Grid view properties dialog at *Magnify/ Overall*. In axonometric view (*Graph type = 3D*), 1/100th of the vertical exaggeration can be given (*Magnify/ Vertical*). The axis scales are plotted with the *Scale* box. When visualising model output grids, dry cells can be coloured differently. It may be useful in an axonometric view to hide cells located on the land-water boundary by checking *Invisible shoreline*. The colour scale, the accompanying legend and the field variable on which it is based can be set in each tab. The next screenshot explains the function four controls.



The graphical elements that are independent from the field variables - *Flow lines*, the *Caption* and the *Base map* - can be set using the three buttons on the right. The *Base map* button is enabled only if the project already contains an imported picture.

The available settings depend on whether the view was initiated from the project or the simulation window, according to the following table:

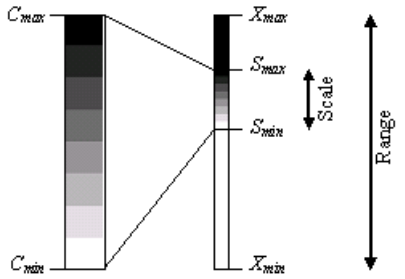
© 2005 Tamás Krámer, www.vit.bme.hu

	View was open from the project window	View was open from the simulation window
Arrows	Not available (N/A)	Based on grids of the velocity and specific discharge
Gridlines	Based on topography grids	Based on any model output grid (except indicator grids)
Flow lines	N/A	Available if flow lines have already been calculated
Land detect	N/A	Wet and dry cells can be coloured differently, but only wet cells can have a gradient colour.
Edit grid	Select the edited grid in the *Edit* list box	N/A

7.2 Colour scale

The four graphical elements (gridlines, pattern, contours, arrows) can be individually switched on to show the variation of a flow variable across the field with gradually changing colours. The colour scale is either a gradual transition between two selected basic colours [C_{min}, C_{max}] or it is one of the predefined colour scales (Rainbow, LR – linear rainbow, BTC – blue-to-cyan, Kelvin, LOCS – linearized optimal color scale and Topography). The advantage of using the LR, BTC, Kelvin and LOCS scales is that they reduce to linear grayscale when printed in black and white.

The scale consists of 16 grades for the gridlines, contours and arrows; 2, 4, 8, 16, 32, 64 or 128 grades for the pattern. The colour scale limits [S_{min}, S_{max}] are set independenly of the value range [X_{min}, X_{max}], so that a smaller subrange can be visualised with high contrast, for example. Go to the grid properties and switch on *Automatically update scale* to stretch the colour scale limits dynamically to the varying value range while the simulation is running. For vector field variables (e.g. velocty, specific discharge, wind), the colour scale reflects the magnitude of the vector.



The pattern of attribute and boundary maps is coloured according to the colour set in the attribute and boundary definition, respectively, and the gradient colour settings are ignored.

The scale limits are set for the grid, not for the grid viewer window, so that two windows displaying the same grid cannot use a different colour scale concurrently. To set the colour scale for a grid, go to the *Grid properties* dialog box of the respective grid either (a) in the grid list of the project/simulation window or (b) in the display properties by right-clicking in the grid viewer window or by selecting **View/View properties** when that window is active then pressing the Grid... button.

When visualising simulation grids, dry cells can be distinguished from wetted or wettable cells by selecting a different colour. Three land detection methods are available:

- Wet vs. dry cells according to the current water surface,
- Water accessible vs. land-type cells according to the attribute definition,
- Flooded vs. never flooded cells, meaningful when wetting and drying is enabled
- All grid cells may be treated without distinction.

© 2005 Tamás Krámer, www.vit.bme.hu

Input grids visualised from the project window do not see any wet-dry information (even when a simulation is running), therefore no land detection is available in those windows.

Legend, axis scale

Check the *Legend* box in the tab of the graphical element to show the legend key to the colour scale. The legend relates the colours to the field variable values if they are gradient coloured otherwise no legend is displayed. Arrow fields express the magnitude also with their size, therefore they may have a legend even if they are uniformly coloured. Set the position of the legends to the left or bottom of the window or to an automatic placement.

The physical size of the flow domain is shown by the axis scale (*Grid view properties* window, *scale* check box). Coordinates are relative to the origin in the bottom left corner of the grid. Absolute coordinates cannot be displayed.

7.3 Caption

A multiline caption may be inserted into grid views to add text to the graphics, such as the title of the simulation. The caption may contain static or dynamic fields that are substituted with the respective value and are updated in the course of the simulation. The recognized codes are the following:

{MT}	Model time (date and time)
{ET}	Elapsed time (date and time)
{MTT}, {MTD}, {ETT}, {ETD}, {ETH}	Model time (time only, date only), Elapsed time (time only, date only, decimal hours only)
{SC}	Simulation step count (incremented by one at each timestep)
{N}	Filename
{FN}	Full filename with folder
{Z,1}	Water surface elevation at boundary 1, [m]
{Q,3}	Discharge at boundary 3, [m^3/s]
{WB,1}	Weir width at boundary 1, [m]
{WZ,1}	Weir crest elevation at boundary 1, [m]

The caption can be aligned horizontally and vertically to the window frame or to the bounding rectangle of the grid.

To turn on caption in grid views, bring up the *Grid view properties* dialog and press *Caption*.

7.4 Base map

A base map laid on the grid facilitates the interpretation of the flow fields. Laying a base map onto the grid in the grid viewer is accomplished by clicking the *Base map...* button in the *Grid view properties* and selecting one of the *Pictures* already imported to the project. The base map is by default drawn on top of the fill, but a different position can be selected from the *Drawing order* listbox. Portions of the base map that extend beyond the grid rectangle can be cropped by checking *Crop to grid*.

Some pictures (such as those exported from Autocad) have a non-transparent background that would obstruct all of the graphics below. To avoid this, check the *No fill* box, so that filled entities in the picture will be drawn with their hollow contour.

© 2005 Tamás Krámer, www.vit.bme.hu

7.5 Flow lines

Flow lines facilitate the visualisation of velocity fields. These lines are locally tangent to the velocity and are computed by the piecewise analytical particle tracking method of Pollock (1988).

- The *pathline* shows the path followed by a particle released from the seeding point.
- The *streakline* connects the particles released continuously from a seeding point and advected by the flow (equivalent to a plume).
- The *time line* connects particles released simultaneously along a curve and advected by the flow (equivalent to a smoke ring).

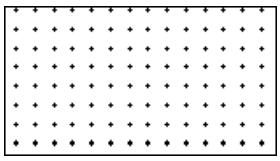
In stationary flow fields, the stream line and the streak line are equivalent.

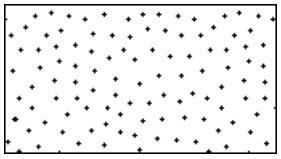
To visualise streamlines in SWAN, you need to define seeding points, compute streamlines using the active velocity field then display streamlines in a grid viewer window.

Define seeding points

Seeding points are defined through a list of points with their I,J grid indexes. The list of points is either (a) generated by SWAN or (b) imported into the project.

(a) Seeding points generated by SWAN cover the whole flow field uniformly in a random fashion or in a regular array. Once the model has been initialised (in order to delimit wet cells) select **Define/ Seed locations...** from the menu. The (mean) horizontal and vertical spacing as well as the maximum number of points must be defined. The generated list of points will be added to the project, under the category *Vector time series* in the project window and can also be used as the pattern of vector field arrows.


Regular


Random

(b) To explicitly define seeding points, import their list as *Vector time series (Cartesian),* with the time information disregarded. This list can be created in Notepad or Excel by entering the fractional I and J index of each seeding point, one point on a line. Use the following formulae to convert physical coordinates (x,y) to grid indexes (I,J) for grid alilgned physical coordinate systems:

$I = (x - x_{left}) / dx,$

$J = (y_{top} - y) / dy,$

where x_{left} and y_{top} are the left and bottom coordinates specified in the *Grid properties*.

Compute flow lines

The flow lines are computed according to the actual velocity field, therefore the simulation must be initialised beforehand. In addition, stationary pathlines require that the simulation has already converged to steady-state. To initiate the calculation of flowlines, choose **Calculate/ flow lines...** from the menu. The *Line type* defines whether only the last trace - or vertex - of the flowlines is advected (= pathline) or all the vertices of the flowlines (=streakline) during a timestep. The third line type, *evenly spaced streamlines*, works somewhat differently and is discussed in the next paragraph. *Time resolution* sets the time interval between the traces of the flow lines: a smaller value gives smoother flow lines at the expense of more computation. However, the accuracy of the individual trace positions is not influenced by the time resolution, just the number of traces. Control the *maximum length* of the flowlines by specifying the upper limit for the number of traces up to 10 000. If the flowlines grow past this length, the oldest traces will be deleted to keep the length at the specified maximum. The seeding points (the initial trace positions) are selected from the list. It is also possible to generate uniform seed locations within the *Flow lines* dialog box. To delete unnecessary seed locations, close the dialog box, go to the project window and delete them from the *Vector time series*.

If you select *evenly spaced streamlines,* the whole field is filled at once by pathlines with a uniform distribution, by freezing the current velocity field with the method of Jobard & Lefer (1997). The *average spacing* (*s*) of the pathlines is controled by the user. With evenly spaced streamlines, the seed

© 2005 Tamás Krámer, www.vit.bme.hu

locations are not given explicitly, they are instead determined automatically with respect to the desired spacing. To bring flowlines closer where the velocity is high, specify a factor for the *smallest spacing*, called *f*. The spacing of the pathlines will vary gradually with the local velocity, between *s* to *f*s.* The evenly spaced streamlines are calculated once the dialog box is closed with the *Add* button. It may take several minutes, especially with high number of grid nodes and small spacings, therefore these pathlines are not updated if the velocity field changes, they must be instead recalculated manually.

Streamlines can be *Added*, *Replaced*, *Deleted* or *Copied* to the clipboard at any time during the simulation.

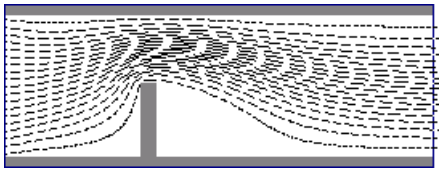
Visualise flow lines

Once calculated, flow lines can be displayed in grid viewer windows of model output, by pressing the *Flow lines...* button in the Grid view properties dialog. The button is grayed if this is an input grid viewer or if no flow lines have been calculated. *Pathlines*, *Time lines* and *Particles* may switched on and edited individually.

The value in the *Skip* field controls the interval for drawing path lines or time levels. Setting *Skip* = 1 will draw all traces and flow lines. To have every 10th pathline or streakline, set *Skip* = 10 under *Pathlines / streaklines*. To display every 10th time line or particle trace, set *Skip*=10 under *Time lines*.

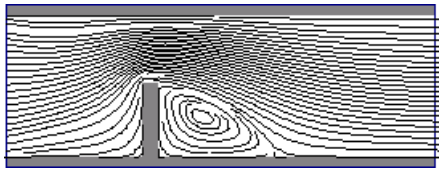
Additionally, pathlines/streaklines can be dashed by checking *Draw with alternating colour*. Length (1) is length (2) are the number of traces to show with *Colour* and *Colour (2)*, respectively. The segment length between two consecutive traces is the distance the trace travelled during the time interval of the flow line, the dashes are therefore proportional to the velocity.

The following examples show how to use flow lines to visualise the east directed flow around a spur dike. The seed points were distributed along the inflow section for the first three cases and near the square corners in the last figure.

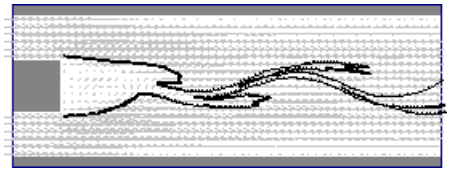


Pathlines

Time lines



Evenly spaced streamlines



Streaklines (with particles)

7.6 Flood maps

By post-processing the flow solution of flood problems, flood dynamics and floodplain vulnerability can be expressed with *flood maps*. The flood maps offered by SWAN show the distribution of the following field variables:

- flood arrival time since the start of the simulation,

© 2005 Tamás Krámer, www.vit.bme.hu

- flood retreat time,
- peak flood level and peak flood depth,
- time when the peak occurred,
- flood residence time.

Before starting the simulation, switch on the calculation of *Flood maps* in the dialog box brought up by the menu command **Define/ Flow...** The flood maps are available for visualisation or copying in the simulation window. To track flood variables, record time series on the flood maps at any number of nodes.

8 Export results

8.1 Export graphics

Contents of a grid viewer window can be exported to file or copied to the clipboard in raster (BMP) and vector (WMF) formats using **Edit/ Copy** or **Edit/ Export...**, respectively. Both file formats are native to Windows and supported by all major office and illustration software. To copy graphics to the clipboard, you can use the keyboard shortcut **Ctrl+Ins**. Data appears on the clipboard in four flavours: *Picture* and *Picture (Enhanced metafile)* are the vector formats, *Bitmap* and *Device independent bitmap* are the raster formats.

SWAN does not provide any printing function. To print graphics, copy the data to a printing-enabled application such as Word. When pasting the graphics to an office application, choose the vector formats for unlimited printing resolution and the raster formats for quick, robust display at screen resolution.

8.2 Record and export time series

Record time series to track the evolution of field variables at discrete nodes.

Recording points

A list of named *Recording points* must first be entered in the dialog **Define/ Recording points...** with the syntax {name$_k$} = {I_k}, {J_k}, e.g.:

```
dam = 12,7
A = 58,23
B = 152,20
```

The indices *I*, *J* refer to grid nodes and can be read on top of the floating *Value* box in a grid viewer window. If the *Value* box is not visible, double-click on any node to display it.

Record time series

To record the history of a field variable, right-click the respective grid in the simulation window and select **Record time series...** at any stage of the simulation. This brings up a dialog with the recording parameters. The time *Interval* is set independently of the simulation timestep. Recording is done in an integral fashion: the element in the time series represents the nodal time-average of that field variable during the interval. To record at a single node, enter its *I*, *J* indices in the *Position* fields and press *OK*. To record at all predefined recording points, close the dialog box by clicking *All points.*

Recorded time series are listed on the bottom of the simulation window. Repeat the procedure above for other field variables to record simultaneously any number of time series.

The time series will be truncated if the simulation is reset which causes the previously recorded data to be discarded. Similarly to simulation field variables, the time series are not saved in the *swn* file and are lost with the closing of the simulation window.

Export time series

Export time series to the clipboard or to an ASCII file with the menu command **Edit/ Copy** and **Edit/**

© 2005 Tamás Krámer, www.vit.bme.hu

Export, respectively, after having selected the time series in the list box. The time series will be copied without any header, consisting of the average values recorded at the given time interval. To export all recorded time series simultaneously preceded by a date-time column, choose **Model I/O/ Copy all time series**.

8.3 Export grid data

To export the numerical contents of a grid to the clipboard or to a file, right-click the grid in the simulation window or the project window and choose **Copy** or **Export** from the context menu. You can also use find these commands in the **Edit** menu. When exporting to a file, several file formats are available. By saving in *text format*, a table of values delimited by tabs is written. The three *Surfer GRD* formats save it for reading into the Surfer program. For attribute maps and indicator maps, the *Surfer BLN* format will replace *Surfer GRD* and will export the boundary of the wet cells for blanking in Surfer. More precisely, the following boundary is generated depending on the grid:

Field variable	**Boundary in BLN file for...**
Attribute map	Cells with wettable attribute
Indicator	Currently wet cells
Indicator (flow)	Cells with wettable attribute
Flood indicator	Cells flooded at some point during the simulation

The *Tecplot PLT* format binds all variables in a single Tecplot data file.

Tecplot and Surfer files are written in terms of physical coordinates, respecting the *Left* and *Bottom* coordinates specified in the grid properties.

8.4 Store flow fields

Store intermediate flow fields for replaying the simulation or for resuming the calculations later.

Output

The parameters controlling the output of flow fields are accessible by selecting **Define/ Flow...** from the menu that brings up the *Flow computation* dialog, then by pressing the *Settings...* button under the *Time stepping* category. Check *Write to*, enter a name for the simulation in the box below and the *Time interval* between two outputs. The simulation must be reset if these output settings are changed. On reset, SWAN will create two entries under *Grid-time series* in the project window: one for the depth and one for the specific discharge vector. These entries are links to external GTS files in the same folder as the SWN project file. To see the details of the external files, look at the properties of the grid-time series (right-click then *Properties...*). The reference contains a relative path, therefore these GTS files must be moved together with the SWN project file.

The depth and the specific discharge fields are sufficient to reconstruct the flow field at the time resolution given in *Time interval*. A trade-off is often necessary between time resolution and the file sizes. In fact GTS files can grow quite large (several hundred megabytes is not uncommon for long unsteady simulations). Nevertheless, these GTS files can later be compressed efficiently for archiving. Similarly to recorded time series, the GTS files are rewound and truncated when the simulation is reset. SWAN will give a warning in such a case, and offer to switch off writing to avoid overwriting the old results (that may be the result of a long simulation).

To back up the latest result of the simulation at regular intervals without keeping the previous results, check the appropriate box under *Automatically export at each interval*. This is especially useful for steady-state simulations, to mitigate the effect of a crashing computer or sudden numerical instability. Saving the *Flow field* saves the fields *h* and *q* to grids which are sufficient to resume the simulation

© 2005 Tamás Krámer, www.vit.bme.hu

from that stage. Saving the *Flood maps* saves the flood field variables (the peak flood level, the flood arrival time etc.). The *Tecplot* format saves to a *flow.plt* (or *flood.plt*) file in the same folder as the SWN file. The Surfer *GRD* format saves

- Flow fields to *h.grd* (depth), *vx.grd*, *vy.grd* (velocity components), *qx.grd*, *qy.grd* (discharge components) and *qsx.grd*, *qsy.grd* (half-staggered discharge components) grid files,
- Flood maps to *t_arriv.grd* (arrival time), *t_resid.grd* (residence time), *t_retr.grd* (retreat time), *t_zmax.grd* (peak time), *zmax.grd* (peak level), *hmax.grd* (peak depth) grid files.

To write the actual solution to GRD and Tecplot files manually, choose **Model IO/ Export flow grids** and **Model IO/ Export flood grids**, which save the above grids in the same folder as the SWN file.

Warning: Keep in mind the following to avoid losing simulation data:

- To preserve the previously computed GTS files, disable flow field writing before resetting (or initialising) a simulation or enter a new simulation name in the *Write to* box.
- Save the SWN project file before closing so that the reference to the GTS files is remembered next time the project is reopened.

Input

By reading the solution of a previous simulation, we can use SWAN to visualise flow fields, convert the results to a different format or use it as the initial condition for another simulation (called hotstarting).

Hotstart

To hot-start with a previous solution, choose **Calculate/ Hotstart...** from the menu, select the *Simulation* in the list box from the grid-time series available in the project file (the Simulation name was given to the stored flow fields in the *Write to* box). Set the timestep to impose as initial condition in the *End* field. The default timestep is the last one in the grid-time series.

It is also possible to overwrite model results with the content of GRD files using **Model IO/ Import flow grids (GRD)**. These solution grid files were previously written either by automatic backup or explicitly using **Model IO/ Export flow grids**. The same function is available for flood grids. Use **Model IO/ Update all fields** after any modification of the solution grids.

Replay

A previous solution can be replayed for the purpose of animation or time series recording. Choose **Calculate/ Replay mode** from the menu, select the *Simulation* from the available grid-time series in the file, the *Start date* and *End date* and the timestep. Initialise the model if not yet done, open any grid viewers and use start, step, pause and reset buttons to navigate in the stored results with the specified timestep. No backward stepping is currently possible, only full rewind (= reset). While in replay mode, all steps can be written to a single Tecplot file (the steps being represented by zones) with **Model IO/ Export replay**. To exit from replay mode, choose **Calculate/ Replay mode** and remove the check from *Enable replay mode*.

9 User interface

9.1 Project window

The software has a familiar, so-called Multiple Document Interface, that is, client windows can be opened and rearranged in the main window and provides standard functionality:

- Familiar menus
- Right-clicking brings up the *context* menu
- Toolbar
- Data exchange through the Windows clipboard

The project window is the main window that opens when a file is opened or created. It lists the items added to the project, and allows their editing.

© 2005 Tamás Krámer, www.vit.bme.hu

The items are listed arranged by type:

- Topography grids
- Attribute maps
- Boundary maps
- Scalar time series
- Vector time series
- Grid-time series
- Pictures

The titles of empty categories do not appear in the project window.

The **Edit** menu (and the context menu) contains the commands available for the selected item.

- *Edit...*
  Opens a grid viewer window for the selected grid which allows the editing.
- *Resolution...*
  Reinterpolates the grid to a different cell resolution. This method may produce crude results, therefore it is recommended that the grids are recreated from the original data.
- *Activate*
  Selects the grid for use in the simulation. Active grids ara marked with the text *(active)*.
- *Cut*
  Moves the selected item to the clipboard then removes it from the project.
- *Copy*
  Copies the item to the clipboard.
- *Paste*
  Insert the clipboard contents to the project. Specify in the dialog boxes the type of item, whether to overwrite the selected item. The pasted item will be placed to the last position in its category.
- *Delete*
  Removes the item from the project. Active grids cannot be deleted. When deleting grid-time series, the corresponding GTS files are deleted only with the user's confirmation.
- *Export...*
  Copies the item to a file.
- *Import...*
  Pastes an item from a file. The procedure is the same as with *Paste*.
- *Properties*
  Edit the object's name and other properties.

Closing the project window is equivalent to closing the file. A confirmation is asked if the project has been changed since the last save.

9.2 Simulation window

By initialising the simulation with the command **Calculate/ Initialise**, the simulation window is opened on the right. The top panel in the window informs about the progress of the simulation. The model output grids and the recorded time series appear below the top panel. The category *Model input grids* lists the active grids from the project window, the *Model result grids* category lists the fields of the model output. If the menu item **View/ Show internal grids** is checked, the internal grids are displayed beneath the output. These contain the grids of internal variables useful for debugging or to inspect intermediate results such as staggered discharge components or the indicator grid. The category *model time series* is present only if time series are being recorded. An entry is added for each recorded variable and each recording point.

Opening the simulation window simultaneously resets the simulation to the initial conditions and enables relevant commands under the **Calculate**:

- *Initialise*
  Opens the project window and initialises the simulation.

© 2005 Tamás Krámer, www.vit.bme.hu

- *Start*
  Launches the simulation (or the replay).
- *Increment*
  Performs one timestep then stops.
- *Stop*
  Pauses the simulation.
- *Reset*
  Resets the simulation to the beginning (or rewinds the replay). All results are overwritten according to the initial conditions!
- *Record time series...*
  Inserts a new time series that tracks the flow variable selected in the simulation window.
- *Flow lines...*
  Controls the calculation of flow lines.
- *Replay*
  Activates the replay mode, where the evolution of the flow variables is read from grid-time series stored on disk.

Closing the simulation window is equivalent to terminating the simulation. Therefore, if the simulation window unnecessarily obstructs the screen, minimise it instead of closing.

9.3 Grid view/edit window

Selecting **Edit/ Edit...** from the project window or **Edit/ View...** from the simulation window or double-clicking while a grid is selected opens a grid viewer window to graphically edit or view the contents of the field variable. Model grids can only be viewed, not edited.

Edit grids

In editing mode, the active node is highlighted with a round cursor. The variable value at the highlighted node is displayed in the *Edit* field of the floating *Value* window. If that window is not visible, double-click on the node or select **View/ Show values** from the menu. To move the cursor, click a node with the mose to make it the active node or use the arrows keys on the keyboard. Holding down the *Shift* key jumps by 10 nodes at once, holding down the *Ctrl* key jumps to the respective end of the row or column. To change the grid value, enter the new value in the *Edit* field of the floating window. This is done like in Excel:

- pressing F2,
- left-clicking in the text box of the *Edit* field, or
- entering the first digit of the new value, or
- pressing Shift-Insert, which copies the clipboard contents to the *Edit* field.

The new value can be accepted by:

- pressing *Enter*, or
- selecting another node with the mouse, or
- moving the cursor with the arrow keys, which at the same time moves the cursor to the next node.

Pressing the *Esc* key in the edit box returns to the original value. Once a value is accepted, it is directly written to the grid contents, so there is no need to save the editing to the original grid. There is no undo facility in SWAN. The grid view is redrawn with the changed value in all windows where it is displayed.

9.4 Model parameters dialog

Model parameters are set by choosing the commands in the **Definitions** menu.

- *Attributes...*: dialog to edit the list of attributes
- *Boundaries...*: dialog to edit the list of boundary conditions

© 2005 Tamás Krámer, www.vit.bme.hu

- *Recording points....:* Enter the name and coordinates of points used to record time series
  Seed locations...: Generate a series of points that cover the flow domain uniformly in a random or regular pattern, used for arrow placement in grid viewers and for seeding flow lines. The generated points are inserted as *Vector time series* in the project window. Available only after model initialisation.
- *Flow...*: Set the solved form of the governing equations, reading and writing of simulation grids, wall slip and calculation of flood grids.
- *Model parameters...*
  Brings up the *Model parameters* dialog, where the start and end time, the timestep, the viewer update frequency, the initial water level and the physical constants can be edited.

The majority of model parameters can be changed on the fly, while the simulation is running, though there are some exceptions.

10 Error messages

- **Too many attributes or boundary conditions.**
  Their maximum number is 255.
- **Cannot delete attribute or boundary, because it is used in the following map: <*name of the map*>.**
  First all instances of the ID must be removed from the attribute or boundary map.
- **Cannot delete attribute or boundary, because at least one is needed.**
  First define the replacement, then delete the old one.
- **Invalid name.**
  The objects' name may contain up to 31 characters.
- **Cannot delete grid because it is used in the calculations.**
  Activate another grid of the same category before attempting to delete the old one
- **Zero curvature prescribed at open boundary.Error: cannot extend in the given driection. Check Q-position in the boundary definition.**
  With the *ZeroCurvature* and *ZeroCurvatureIfOut* switches, the boundary values are extrapolated from the interior. The interior is identified as the side opposite to the one where the discharge is prescribed. E.g. if Q is prescribed on the East side, the boundary value is extrapolated from the two West neighbours. It is therefore necessary to give the position of the boundary when it is the water level which is prescribed with the *ZeroCurvature*.
- **Grid-time series data file is missing.**
  The data of grid-time series is stored in external GTS files. These must be moved with the SWN project file.
- **A boundary condition prescribes time series instead of a value in the Auxiliary data (e.g. WeirLevel, WeirWidth), but the time series is not found or of the wrong type.**
  SWAN has found that a weir-type discharge boundary is specified with unsteady width or crest elevation because the name of a time series was assigned to WeirLevel or WeirWidth. Make sure that the time series really exists in the project window or if the value was meant to be a numeric value, but mistyped.

11 More information

Inquire here for more information about the SWAN program:

Tamás Krámer
Budapest University of Technology and Economics
Department of Hydraulic and Water Resources Engineering
H-1111 Budapest, Műegyetem rkp 3., K mf. 4.
Tel: +36 1463-1495, email: kramer@vit.bme.hu
Web: www.vit.bme.hu

© 2005 Tamás Krámer, www.vit.bme.hu

12 References

Burchard, H.: Presentation of a new numerical model for turbulent flow in estuaries and wadden seas. *Proc. Hydroinformatics '98*, Copenhagen, A.A. Balkema, Rotterdam, 1998.

Józsa J.; Krámer T.; Bakonyi P.: Ártéri öblözetek töltésszakadást követő elöntési folyamatainak modellezése II.: Az ártéri modell. *Hidrológiai Közlöny*, 1999; **4**: 234–240.

Jobard, B.; Lefer, W.: Creating evenly spaced streamlines of arbitrary density, *Proc. Visualization in Scientific Computing '*97. pp 45–55, Boulogne-sur-Mer, France, 1997

Kozák M.: A szabadfelszínű nem-permanens áramlások számítása. Akadémiai Kiadó, Budapest, 1977.

Pollock, D.W.: Semianalytical computation of path lines for finite-difference models. *Ground Water*, 1988; **26**: 743–750

© 2005 Tamás Krámer, www.vit.bme.hu